

Máster Avanzado de Programación en Python para Hacking, BigData y Machine Learning

Programación Avanzada Python

LECCIÓN 7

Trabajando con BBDD

ÍNDICE

Introducción

Objetivos

SQL

Sintaxis SQL

SQLite Python

Trabajando con SQL

INTRODUCCIÓN

En este séptimo capítulo nos adentraremos en el mundo de las BBDD. Primero veremos los fundamentos necesarios del lenguaje de consultas de base de datos SQL y por último como trabajar de forma práctica con SQL y Python mediante SQLite.

OBJETIVOS

Al finalizar esta lección serás capaz de:

- 1 Conocer los fundamentos de BBDD.
- 2 Trabajar con SQL
- 3 Combinar Python y SQL
- 4 Crear tu propia BBDD

SQL

SQL es el lenguaje estándar para tratar con bases de datos relacionales, se puede utilizar para insertar, buscar, actualizar y eliminar registros de bases de datos y puede realizar muchas otras operaciones, incluida la optimización y el mantenimiento de bases de datos.

¿Para qué se utiliza SQL?

- Ayuda a los usuarios a acceder a los datos en el sistema RSGBD.
- Ayuda a describir los datos.
- Permite definir los datos en una base de datos y manipular esos datos específicos.
- Con la ayuda de SQL, puede crear y eliminar bases de datos y tablas.
- SQL nos ofrece utilizar la función en una base de datos, crear una vista y un procedimiento almacenado.
- Puede establecer permisos en tablas, procedimientos y vistas.

Tipos de declaraciones SQL

A continuación, se muestran cinco tipos de consultas SQL de uso generalizado.

- Lenguaje de definición de datos (DDL)
- Lenguaje de manipulación de datos (DML)
- Lenguaje de control de datos (DCL)
- Lenguaje de control de transacciones (TCL)
- Lenguaje de consulta de datos (DQL)

Lista de comandos SQL

A continuación, se muestra una lista de algunos de los **comandos SQL** más utilizados:

- **Create:** define el esquema de la estructura de la base de datos
- **Insert:** inserta datos en la fila de una tabla
- **Update:** actualiza los datos en una base de datos
- **Delete:** elimina una o más filas de una tabla
- **Select:** selecciona el atributo según la condición descrita por la cláusula WHERE
- **Drop:** elimina tablas y bases de datos

Sintaxis

Todas las declaraciones SQL comienzan con cualquiera de las palabras clave como SELECT, INSERT, UPDATE, DELETE, ALTER, DROP, CREATE, USE, SHOW y todas las declaraciones terminan con un punto y coma (;).

El punto más importante que debemos de tener en cuenta aquí es que SQL no distingue entre mayúsculas y minúsculas, lo que significa que SELECT y select tienen el mismo significado en las declaraciones. Mientras que MySQL hace la diferencia en los nombres de las tablas. Por lo tanto, si estamos trabajando con MySQL, debe proporcionar los nombres de las tablas tal y como existen en la base de datos.

Principales sintaxis SQL

Instrucción SQL SELECT

```
SELECT column1, column2....columnN  
FROM table_name;
```

Cláusula SQL DISTINCT

```
SELECT DISTINCT column1, column2....columnN  
FROM table_name;
```

Cláusula WHERE de SQL

```
SELECT column1, column2....columnN  
FROM table_name  
WHERE CONDITION;
```

Cláusula Y / O de SQL

```
SELECT column1, column2....columnN  
FROM table_name  
WHERE CONDITION-1 {AND|OR} CONDITION-2;
```

Cláusula SQL IN

```
SELECT column1, column2....columnN  
FROM table_name  
WHERE column_name IN (val-1, val-2,...val-N);
```

Cláusula SQL BETWEEN

```
SELECT column1, column2....columnN  
FROM table_name  
WHERE column_name BETWEEN val-1 AND val-2;
```

Cláusula LIKE de SQL

```
SELECT column1, column2....columnN  
FROM table_name  
WHERE column_name LIKE { PATTERN };
```

Cláusula ORDER BY de SQL

```
SELECT column1, column2....columnN  
FROM table_name  
WHERE CONDITION  
ORDER BY column_name {ASC|DESC};
```

Cláusula SQL GROUP BY

```
SELECT SUM(column_name)  
FROM table_name  
WHERE CONDITION  
GROUP BY column_name;
```

Cláusula SQL COUNT

```
SELECT COUNT(column_name)  
FROM table_name  
WHERE CONDITION;
```

Tipos de datos

El tipo de datos SQL es un atributo que especifica el tipo de datos de cualquier objeto. Cada columna, variable y expresión tiene un tipo de datos relacionado en SQL. Puede utilizar estos tipos de datos al crear sus tablas. Puede elegir un tipo de datos para una columna de tabla según sus necesidades.

Tipos de datos numéricos exactos

TIPO DE DATOS	DESDE	A
<u>bigint</u>	-9,223,372,036,854,775,808	9,223,372,036,854,775,807
int	-2,147,483,648	2,147,483,647
<u>smallint</u>	-32,768	32,767
<u>tinyint</u>	0	255
bit	0	1
decimal	$-10^{38} + 1$	$10^{38} - 1$
numeric	$-10^{38} + 1$	$10^{38} - 1$
money	-922,337,203,685,477.5808	+922,337,203,685,477.5807
<u>smallmoney</u>	-214,748.3648	+214,748.3647

Tipos de datos numéricos aproximados

TIPO DE DATOS	DESDE	A
float	-1.79E + 308	1.79E + 308
real	-3.40E + 38	3.40E + 38

Tipos de datos fecha y hora

TIPO DE DATOS	DESDE	A
datetime	1 de <u>enero</u> de 1753	31 de <u>diciembre</u> de 9999
<u>smalldatetime</u>	1 de <u>enero</u> de 1900	6 de <u>junio</u> de 2079
date	Almacena una fecha como el 30 de junio de 1991	
hour	Almacena una hora del día como las 12:30 p.m.	

Tipos de datos de cadenas de caracteres

TIPO DE DATOS y descripción	
1	<u>char</u> Longitud máxima de 8.000 caracteres (caracteres no Unicode de longitud fija)
2	<u>varchar</u> Máximo de 8.000 caracteres (datos no Unicode de longitud variable).
3	<u>varchar (máximo)</u> Longitud máxima de $2E + 31$ caracteres, datos no Unicode de longitud variable (solo SQL Server 2005).
4	<u>text</u> Datos no Unicode de longitud variable con una longitud máxima de 2.147.483.647 caracteres.

Operadores

Un operador es una palabra reservada o un carácter que se utiliza principalmente en la cláusula WHERE de una declaración SQL para realizar operaciones, como comparaciones y operaciones aritméticas. Estos operadores se utilizan para especificar condiciones en una declaración SQL y para servir como conjunciones para múltiples condiciones en una declaración

- Operadores aritméticos
- Operadores de comparación
- Operadores lógicos
- Operadores utilizados para negar condiciones

Operadores aritméticos SQL

Operador	Descripción	Ejemplo
+	Agrega valores a ambos lados del operador.	a + b dará 30
-	Resta el operando de la derecha del operando de la izquierda.	a - b dará -10
*	Multiplica los valores a ambos lados del operador.	a * b dará 200
/	Divide el operando de la izquierda por el operando de la derecha.	b / a dará 2
%	Divide el operando de la izquierda por el operando de la derecha y devuelve el resto.	b% a dará 0

Operadores comparación SQL

Operador	Descripción	Ejemplo
=	Comprueba si los valores de dos operandos son iguales o no, si es así, la condición se convierte en verdadera.	(a = b) no es cierto.
!=	Comprueba si los valores de dos operandos son iguales o no, si los valores no son iguales, la condición se convierte en verdadera.	(a != b) es cierto.
<>	Comprueba si los valores de dos operandos son iguales o no, si los valores no son iguales, la condición se convierte en verdadera.	(a <> b) es cierto.
>	Comprueba si el valor del operando izquierdo es mayor que el valor del operando derecho, si es así, la condición se convierte en verdadera.	(a > b) no es cierto.
<	Comprueba si el valor del operando izquierdo es menor que el valor del operando derecho, si es así, la condición se convierte en verdadera.	(a < b) es cierto.
> =	Comprueba si el valor del operando izquierdo es mayor o igual que el valor del operando derecho, si es así, la condición se convierte en verdadera.	(a > = b) no es cierto.
< =	Comprueba si el valor del operando izquierdo es menor o igual que el valor del operando derecho, si es así, la condición se convierte en verdadera.	(a < = b) es cierto.
! <	Comprueba si el valor del operando izquierdo no es menor que el valor del operando derecho, si es así, la condición se convierte en verdadera.	(a ! < b) es falso.
! >	Comprueba si el valor del operando izquierdo no es mayor que el valor del operando derecho; si es así, la condición se convierte en verdadera.	(a ! > b) es cierto.

SQLITE PYTHON

SQLite es una biblioteca de software que proporciona un sistema de gestión de bases de datos relacional. Lite en SQLite significa ligero en términos de configuración, administración de bases de datos y recursos necesarios.

SQLite tiene las siguientes características: autónomo, sin servidor, sin configuración, transaccional.

CONCLUSIONES



1

SQL es el **lenguaje** estándar para tratar con **bases de datos relacionales**. SQL se puede utilizar para insertar, buscar, actualizar y eliminar registros de bases de datos.

2

Todas las **declaraciones SQL** comienzan con cualquiera de las palabras clave como **SELECT**, **INSERT**, **UPDATE**, **DELETE**, **ALTER**, **DROP**, **CREATE**, **USE**, **SHOW** y todas las declaraciones terminan con un punto y coma (;).

3

SQLite es una biblioteca de software que proporciona un sistema de gestión de bases de datos relacional. Lite en SQLite significa ligero en términos de configuración, administración de bases de datos y recursos necesarios.

MUCHAS GRACIAS POR SU ATENCIÓN



rsanchezi@grupomainjobs.com



Rubén Sánchez Iruela

[linkedin.com/in/ruben-sanchez-iruela-8156799a](https://www.linkedin.com/in/ruben-sanchez-iruela-8156799a)



twitter.com/eiposgrados



facebook.com/eiposgrados



instagram.com/eiposgrados